An improved chaotic optimization algorithm using a new global locally averaged strategy

Tayeb Hamaizia, René Lozi

Published in
Journal of Nonlinear Systems and Applications
Vol. 3, Number 3-4, (2012) pp. 58-63

http://www.watam.org/JNSA/
http://jnsaonline.watsci.org/abstract_pdf/2012v3/v3n2-pdf/1.pdf

AN IMPROVED CHAOTIC OPTIMIZATION ALGORITHM USING A NEW GLOBAL LOCALLY AVERAGED STRATEGY

Tayeb Hamaizia* and René Lozi† [1]

Abstract. Recently chaotic optimization algorithms as an emergent method of global optimization have attracted much attention in engineering applications. Their good performances have been emphasized [1, 2, 3, 12, 13]. In the frame of evolutionary algorithms, the use of chaotic sequences instead of random ones has been introduced by Caponetto and al. [4]. Since their original work, the literature on chaotic optimization is flourishing. They are used in the scope of tuning method for determining the parameters of PID control for an automatic regulator voltage, or in order to solve economic dispatch problems, or also for engineering design optimization and in many others physical, economical and biological problems. Different chaotic mapping have been considered, combined with several working strategies. The assessments of the algorithms have been done with respect to numerous objective functions in 1, 2 or 3-dimension. In this paper we present an improvement of the COLM (Chaotic Optimization based on Lozi Map) presented in [1], which is based on a new global locally averaged strategy. The simulation results are done in 2 and 3-dimension. In 2-dimension the objective function possessing hundreds of local minima is used, in order to test this new method vs the previous one in very tough conditions. In 3-dimension both Griewank and Rosenbrock objective functions are tested. We emphasize improvement of the numerical optimization results.

Keywords. chaotic optimization, global optimization, chaotic mapping.

1 Introduction

Chaos theory (the term chaos was coined par Li and Yorke [5]) is recognized as very useful in many engineering applications. An essential feature of chaotic systems is sensitive dependence on initial condition, (i.e. small changes in the parameters or the starting values for the data lead to drastically different future behaviors). Details about analysis of chaotic behavior can be found in [5, 6, 7, 8, 9]. The application of chaotic sequences can be an interesting alternative to provide the search diversity in an optimization procedure. Due to the non-repetition of chaos, it can carry out overall searches at higher speeds than stochastic

[1] Department of Mathematics, University of Constantine, (25000) Algeria, and Laboratoire J.A. Dieudonné, UMR CNRS n° 7351 Université de Nice Sophia-Antipolis, Parc Valrose 06108 NICE Cedex 02, France.
E-mail: el.tayyeb@umc.edu.dz
†Laboratoire J.A. Dieudonné, UMR CNRS n° 7351 Université de Nice Sophia-Antipolis, Parc Valrose 06108 NICE Cedex 02, France. E-mail: rlozi@unice.fr

ergodic searches that depend on probabilities. A novel chaotic approach is proposed in [1] based on Lozi map [6] which is piecewise linear simplification of the Hénon map [10] and it admits strange attractors.

It is given by

$$\begin{cases} y_1(k) = 1 - a\left|y_1(k-1)\right| + by(k-1) \\ y(k) = y_1(k-1) \end{cases} \qquad (1)$$

where k is the iteration number. In this work, the values of y are normalized in the range [0,1] to each decision variable in 2-dimensional space of optimization problem.

This transformation is given by

$$z(k) = \frac{\left(y(k) - \alpha\right)}{\beta - \alpha} \qquad (2)$$

where $y \in \left[-0.6418, 0.6716\right]$ and $\left[\alpha, \beta\right] = \left[-0.6418, 0.6716\right]$. The parameters used in this work are $a = 1.7$ and $b = 0.5$. Numerical computation leads to the density $d(s)$ of iterated values of $y(k)$ displayed on Fig. 1. In this figure, the density is normalized to 1 over the whole interval [0, 1] i.e.

$$\int_0^1 d(s)ds = 1$$

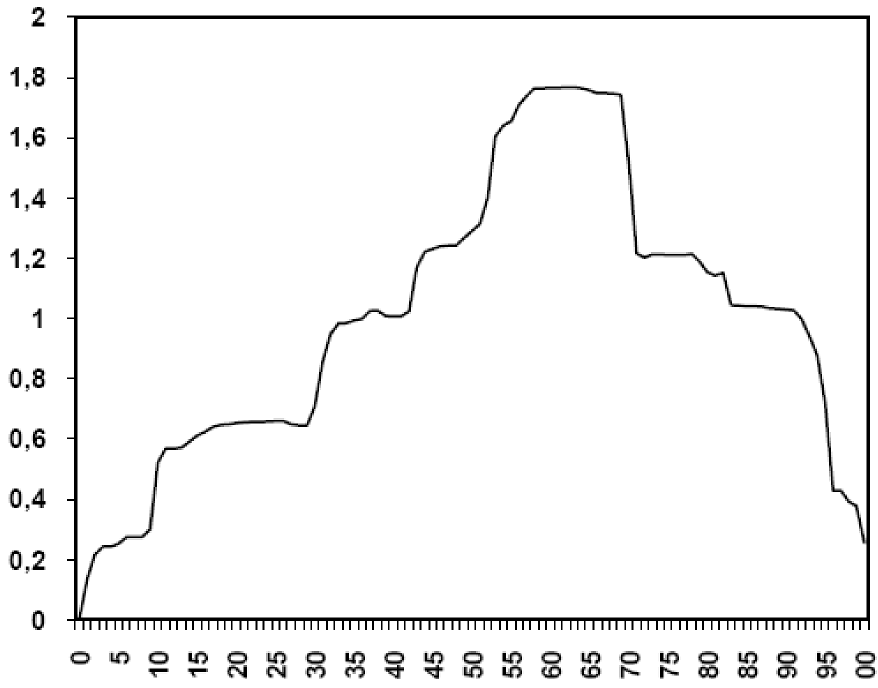

Figure 1: density of iterated values of $y(k)$ of equation (1) over the interval [0, 1] split in 100 boxes for 10,000,000,000 iterated values.

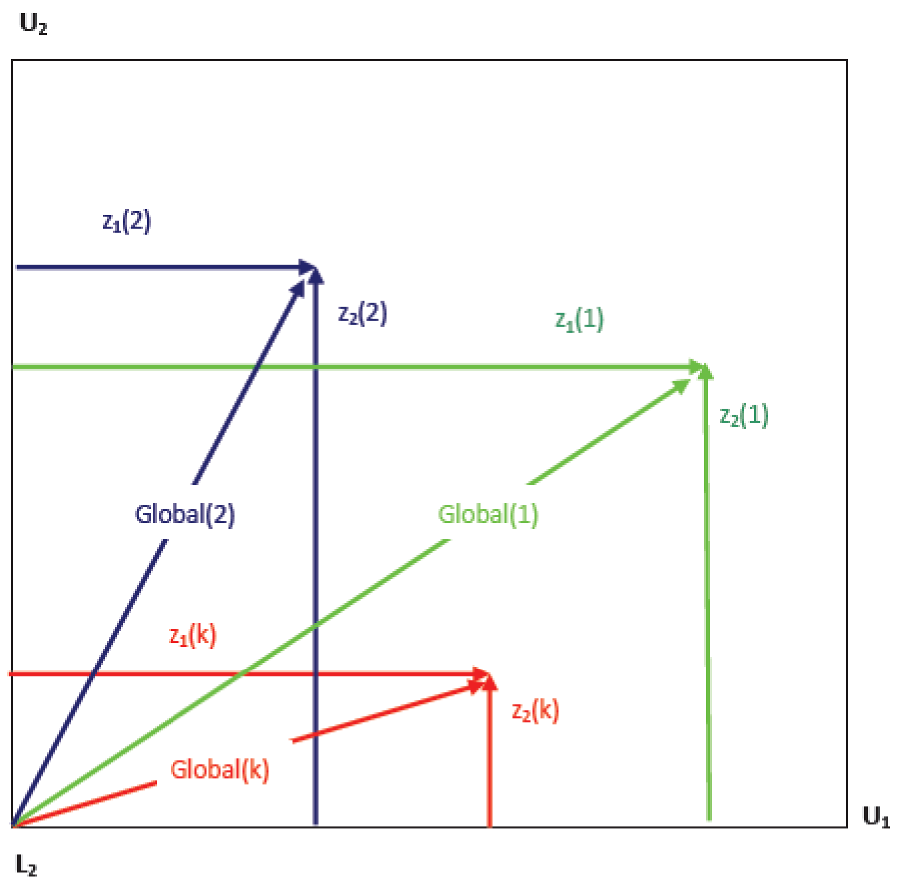

Figure 2: Scheme of global search in COLM. The space of variable is randomly explored by tossing random numbers for every variable.

2 The COLM Algorithm

In this section we briefly recall the algorithm of COLM introduced by L. S. Coelho [1]. There are three steps in this algorithm. In the first one the variables and the initial conditions are initialized. The second step refers to the chaotic global search. It is mainly this second step which will be modified in our method. The last one deals with a local search in order to refine the result of the global search. As this step will be leaved unchanged we will present it in the next section. The chaotic search procedure (step 1 and 2) can be illustrated as follows:

The Figure 2 displays the representation of the chaotic search when the objective function is in dimension 2.

Algorithm 1: COLM

-**Step 1** : Initialize the number M_g, M_l of chaotic search and initialization of variables and initial conditions Set k=1, y (0), $y_1(0)$,$a = 1.7$ and $b = 0.5$ of Lozi map. Set the initial best objective function $\bar{f} = +\infty$

-**Step 2: algorithm of chaotic global search:**

while $k \leq M_g$ **do**
  $x_i(k) = L_i + z_i(k).(U_i - L_i)$
  if $f(X(k)) < \bar{f}$ **then**
    $\bar{X} = X(k); \bar{f} = f(x(k))$
  end if
  $k = k + 1$
end while

3 Improved COLM Method

The COLM method [14, 15] is then improved by locally averaging the global search, doing few steps of chaotic local search around every point obtained by the chaotic series.

Heuristics: the global locally averaged strategy of Improved COLM leads to better results than COLM as shown on Fig. 3. In this figure only three global search results are displayed $x_1(k)$, $x_2(k)$, $x_3(k)$ with

$$f\left(x_2(k)\right) < f\left(x_3(k)\right) < f\left(x_1(k)\right) \quad (3)$$

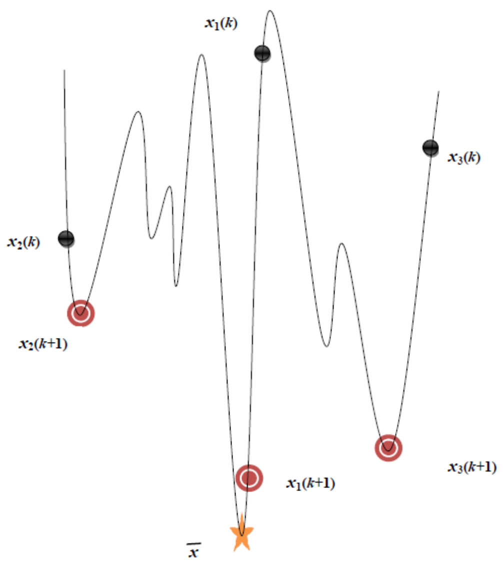

Figure 3: Heuristics of the global locally-averaged strategy.

The local search following global one starts from the best global result $x_2(k)$ (from (3)) and gives $x_2(k+1)$. Instead the local-global search around $x_1(k)$, $x_2(k)$, and $x_3(k)$ leads to $x_1(k+1)$, $x_2(k+1)$, $x_3(k+1)$ which verify

$$f\left(x_1(k+1)\right) < f\left(x_3(k+1)\right) < f\left(x_2(k+1)\right) \tag{4}$$

The local search following the local-global one starts now from the best globally averaged result $x_1(k+1)$ (from(4)) and leads to $\bar{x}$.

$$f(\bar{x}) < f\left(x_1(k+1)\right) \tag{5}$$

Figure 4 shows the Local refinement of each step of the global search. During the chaotic local search, the step λ is an important parameter in convergence behavior of optimization. Hence, two different values of λ are successively employed during the local search. We call this method ICOLM (improved COLM). Many unconstrained optimization problems with continuous variables can be formulated as the following functional optimization problem. Find *X* to minimize *f(X)*, *X* = [x_1, x_2, …,x_n] Subject to $x_i \in \left[L_i, U_i\right]$.

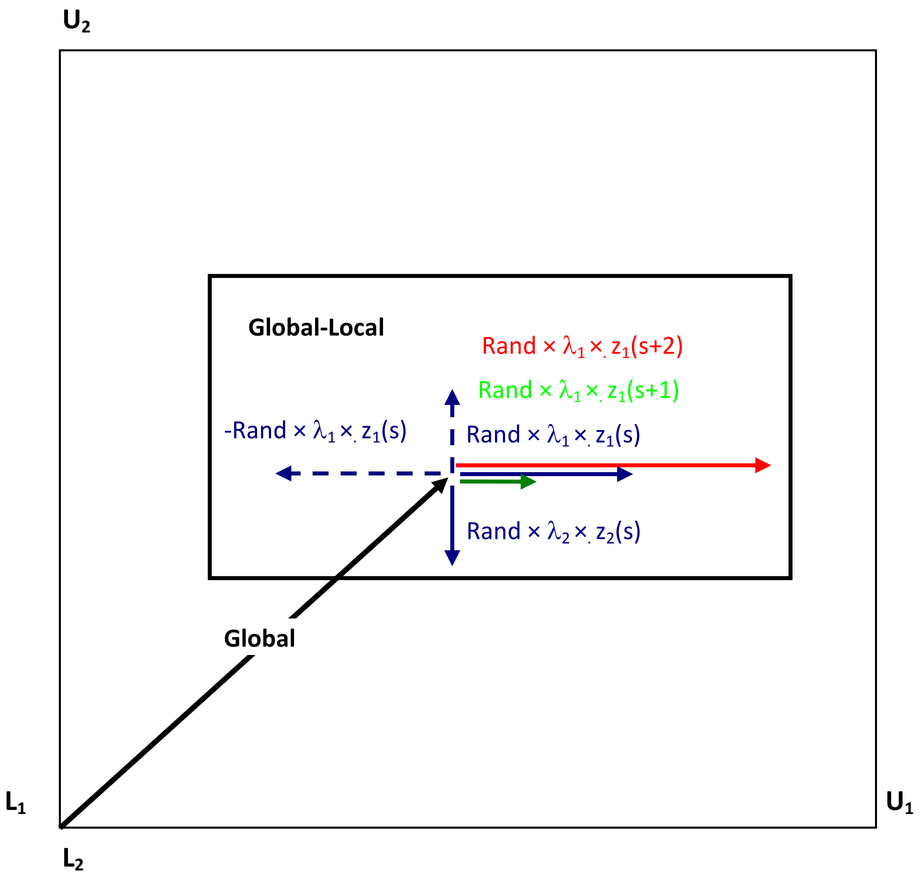

Figure 4: Local refinement of each step of the global search involving both random and chaotic numbers.

Where f is the objective function, and X is the decision solution vector consisting of n variables $x_i \in R^n$ bounded by lower (L_i) and upper limits (U_i). The ICOLM can be illustrated as follows:

Inputs:

M_g : max number of iterations of chaotic Global search.

M_l : max number of iterations of chaotic Local search.

M_{gl_1} : max number of iter of chaotic Local search in Global search.

M_{gl_2} : max number of iter of chaotic Local search in Global search.

$M_g \times \left(M_{gl_1} + M_{gl_2}\right) + M_l$: stopping criterion of chaotic optimization method in iter.

λ_{gl_1} : step size in first global-local search.

λ_{gl_2} : step size in second global-local search.

λ step size in chaotic local search.

Outputs:

$\bar{X}$: best solution from current run of chaotic search.

$\bar{f}$: best objective function (minimization problem).

4 A tough 2-D objective function

In order to test this new method *vs*. the previous one in very tough conditions the simulation results are done with the following 2-D objective function possessing hundreds of local minima: The function *f* which is very complex has several local minima.

$$f_1 = x_1^4 - 7x_1^2 - 3x_1 + x_2^4 - 9x_2^2 - 5x_2 + 11x_1^2x_2^2 + 99\sin(71x_1) + 137\sin(97x_1x_2) + 131\sin(51x_2) \quad (6)$$

We test ICOLM on the search domain: $-10 \le x_i \le 10,\ i = 1,2$. The essential feature of this benchmark function is that location of minima is not symmetric (see Fig. 5, 6).

Algorithm 2: ICOLM

-**Step 1**:Initialize the number M_g, M_{gl1},M_{gl2}, M_l of chaotic search and initialization of variables and initial conditions Set k=1, y (0), $y_1(0)$,$a = 1.7$ and $b = 0.5$ of Lozi map. Set the initial best objective function $\bar{f} = +\infty$

-**Step 2: algorithm of chaotic global search:**

while $k \leq M_g$ **do**
  $x_i(k) = L_i + z_i(k).(U_i - L_i)$
  if $f(X(k)) < \bar{f}$ **then**
    $\bar{X} = X(k); \bar{f} = f(x(k))$
  end if
  -**Step 2-1: sub algorithm of chaotic local search:**
  while $j \leq M_{gl1}$ **do**
    for $i = 0$ **to** n **do**
      if $r \leq 0.5$ **then**
        $x_i(j) = \bar{x}_i + \lambda_{gl1} z_i(j).|(U_i - L_i)|$
      else
        $x_i(j) = \bar{x}_i - \lambda_{gl1} z_i(j).|(U_i - L_i)|$
      end if
    end for
    if $f(X(j)) < \bar{f}$ **then**
      $\bar{X} = X(j); \bar{f} = f(x(j))$
    end if
    $j = j + 1$
  end while
  -**Step 2-2: sub algorithm of chaotic local search:**
  while $s \leq M_{gl2}$ **do**
    for $i = 0$ **to** n **do**
      if $r \leq 0.5$ **then**
        $x_i(s) = \bar{x}_i + \lambda_{gl2} z_i(s).|(U_i - L_i)|$
      else
        $x_i(s) = \bar{x}_i - \lambda_{gl2} z_i(s).|(U_i - L_i)|$
      end if
    end for
    if $f(X(s)) < \bar{f}$ **then**
      $\bar{X} = X(s); \bar{f} = f(x(s))$
    end if
    $s = s + 1$
  end while
  $k = k + 1$
end while

-Step 3: algorithm of chaotic local search:

while $k \le M_g \times (M_{gl1} + M_{gl2}) + M_l$ **do**
  for $i = 0$ to n **do**
    if $r \le 0.5$ **then**
      $x_i(k) = \bar{x_i} + \lambda z_i(k).|(U_i - L_i)|$
    else
      $x_i(k) = \bar{x_i} - \lambda z_i(k).|(U_i - L_i)|$
    end if
  end for
  if $f(X(k)) < \bar{f}$ **then**
    $\bar{X} = X(k); \bar{f} = f(x(k))$
  end if
  $k = k + 1$
end while

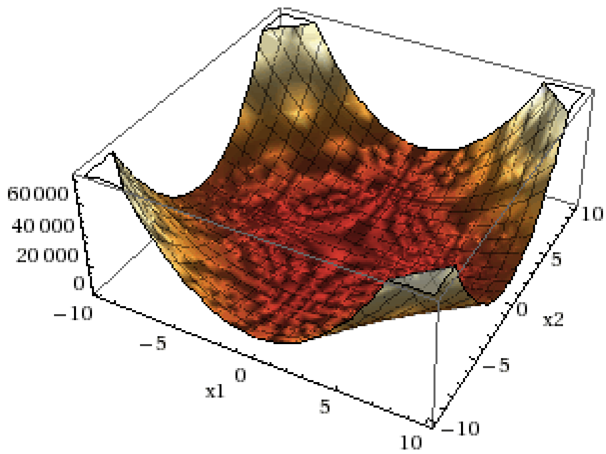

Figure 5: plot of test function f_1 used in this study on the search domain $-10 \le x_i \le 10$, $i = 1, 2$.

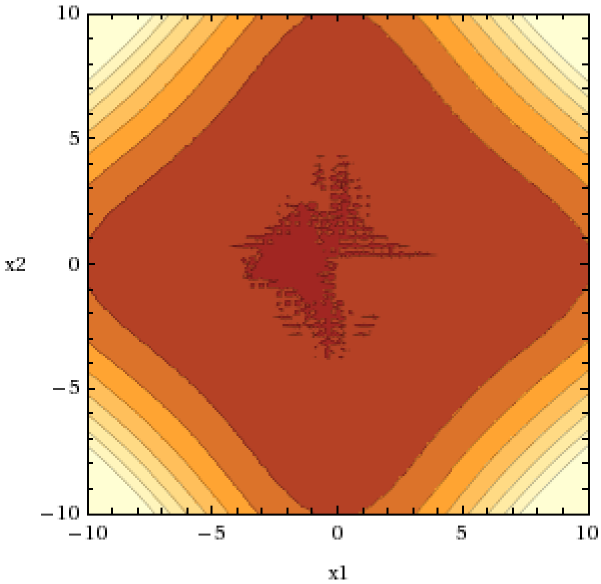

Figure 6: Position of the minima of f_1 in the search domain.

We display few of the results we have obtained showing the better optimization results obtained by this new method. In each case study, 48 independent runs were made for each of both the COLM and ICOLM methods involving 48 different initial trial conditions *$y_1(0)$, $y(0)$* (parameters of Lozi map).

For all studied cases, the four configurations, numbered from IC1 to IC4 and C1 to C4, that are used are presented in Tab. 1. The locally averaged strategy of ICOLM is illustrated on Fig. 7 on which the result of every step 2-2 is plotted.

Table 1: The set of parameters values for every run on the benchmark suite defined in Sec. 4 and 5.

	λ	$\lambda_{M_{gl1}}$	$\lambda_{M_{gl2}}$	M_g	M_l	M_{gl1}	M_{gl2}
IC1	0.001	0.04	0.01	6	50	2	2
IC2	0.01	0.04	0.01	10	50	2	2
IC3	0.1	0.04	0.01	10	50	2	2
IC4	0.1	0.04	0.01	100	50	5	5
C1	0.001			24	50		
C2	0.01			40	50		
C3	0.1			40	50		
C4	0.1			1000	50		

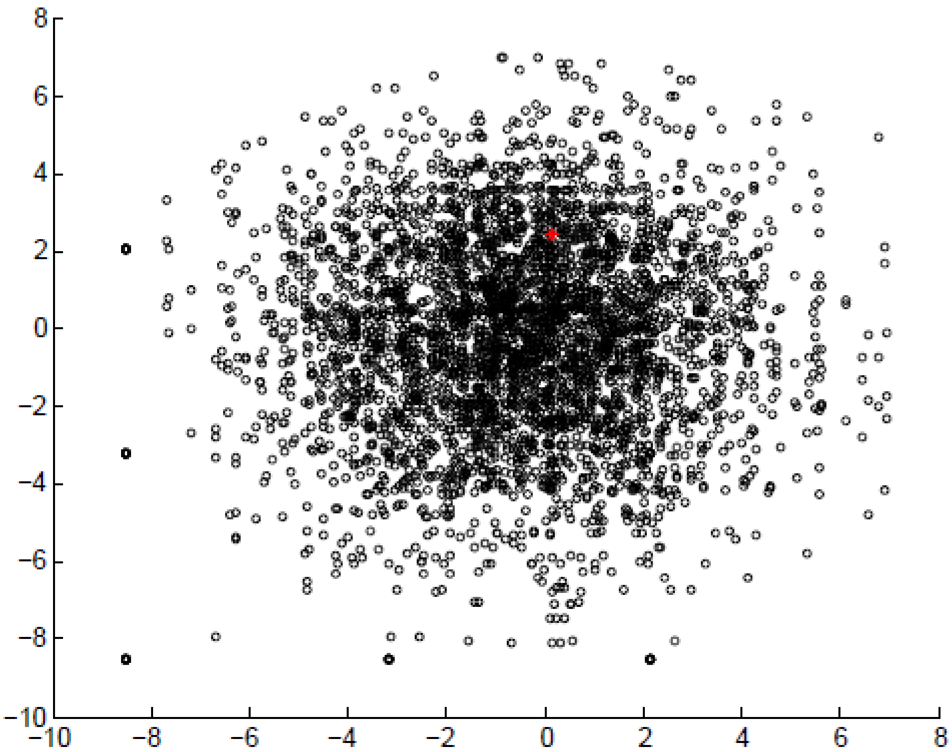

Figure 7: Locally-averaged strategy of chaotic search. Results of every Step 2-2 for *f_1*.

The numerical results are given on table 2. A magnification of f_1 and the position of the minima in the region $0.4 \le x_1 \le 0.8$, $0.3 \le x_2 \le 0.6$ is displayed on Figs. 8, 9.

Table 2: Comparison of algorithms COLM and ICOLM for f_1.

algo	Best value	Mean value	Std.Dev	$\begin{pmatrix}\bar{x}\\ \bar{y}\end{pmatrix}$
New Function				
IC1	-382.3917	-355.8410	13.6581	$\begin{pmatrix}-2.7686\\ -0.4055\end{pmatrix}$
C1	-371.0150	-368.5212	11.1135	$\begin{pmatrix}0.3105\\ 0.2442\end{pmatrix}$
IC2	-392.5400	-365.7837	14.0615	$\begin{pmatrix}-0.7327\\ 1.3203\end{pmatrix}$
C2	-367.3930	-358.0622	7.5935	$\begin{pmatrix}-2.6200\\ 1.5648\end{pmatrix}$
IC3	-393.3134	-379.6872	8.8797	$\begin{pmatrix}-1.9677\\ -1.9982\end{pmatrix}$
C3	-379.0027	-371.5180	5.1866	$\begin{pmatrix}-4.6404\\ -3.0760\end{pmatrix}$
IC4	-395.8441	-387.3368	9.0825	$\begin{pmatrix}-1.6161\\ -2.2475\end{pmatrix}$
C4	-382.7108	-379.7557	1.7817	$\begin{pmatrix}-5.8930\\ 2.9309\end{pmatrix}$

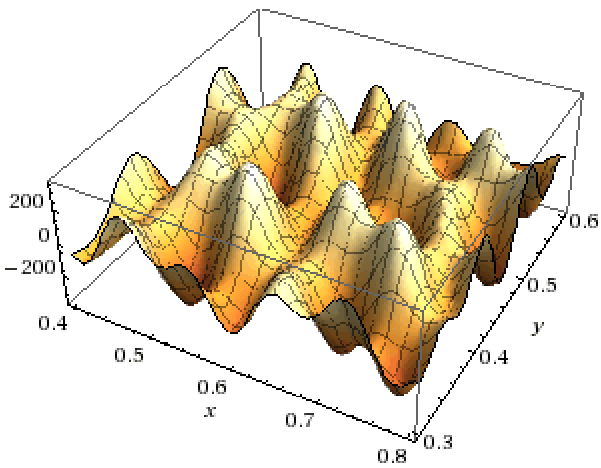

Figure 8: magnification of Fig. 5.

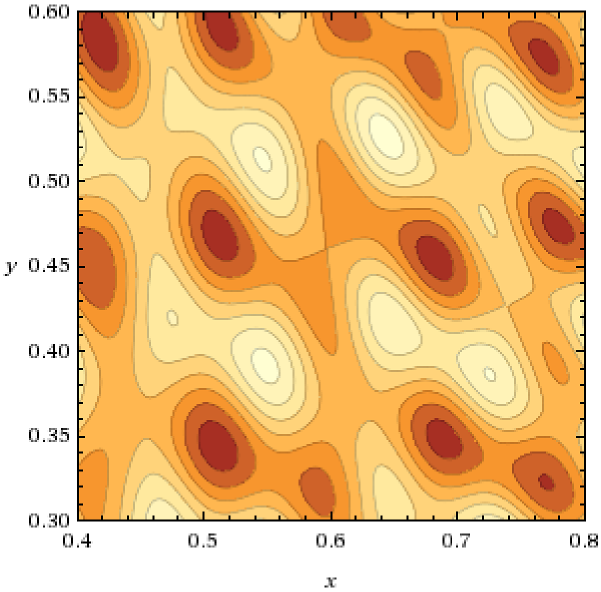

Figure 9: magnification of Fig. 6.

5 3-D Numerical results

ICOLM is again tested in dimension 3, with both following test functions in order to clarify its efficiency.

$$\textit{Griewank:}\quad f_2 = 1 + \sum_{i=1}^{3} \frac{x_i^2}{4000} + \prod_{1}^{3} \cos \frac{x_i}{\sqrt{i}} \qquad (7)$$

in the search domain: $-500 \leq x_i \leq 500$, $i = 1,2,3$.

$$\textit{Rosenbrock:}\quad f_3 = 1 + \sum_{i=1}^{2} 100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2 \qquad (8)$$

in the search domain: $-2.048 \leq x_i \leq 2.048$, $i = 1,2$.

Griewank’s function has many irregularities but it has only one unique global minimum (See Fig. 10). The Rosenbrock’s function is a non-convex function or Rosenbrock's banana function (See Fig 11). The global minimum is inside a long, narrow, parabolic shaped flat valley. To find the valley is trivial. To converge to the global minimum, however, is difficult.

The same cases described in Table 1 are used. The results are given in Table 3 for Grienwank's function and in Table 4 for Rosenbrock's function.

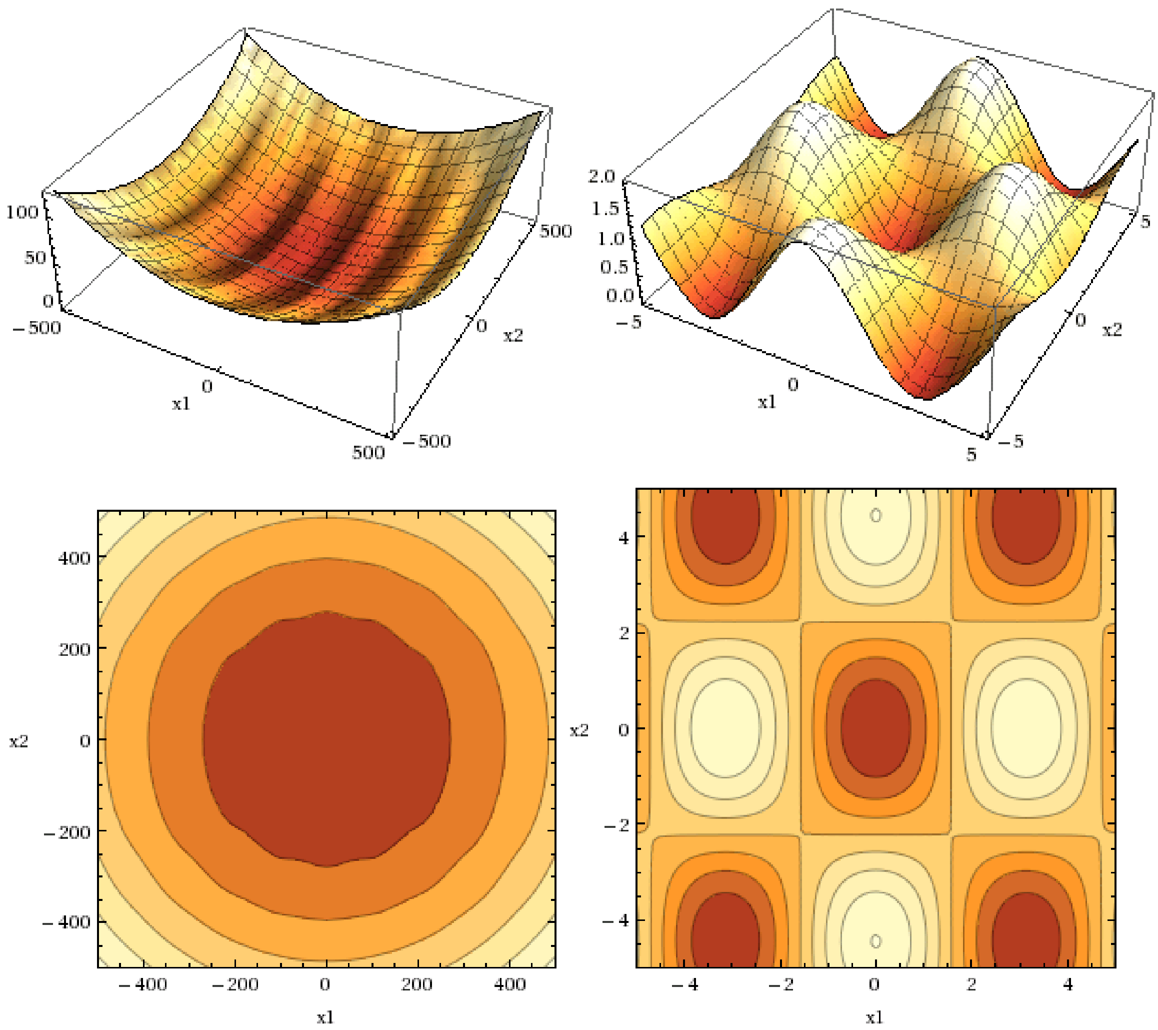

Figure 10: plot of test Grienwank's function used in this study.

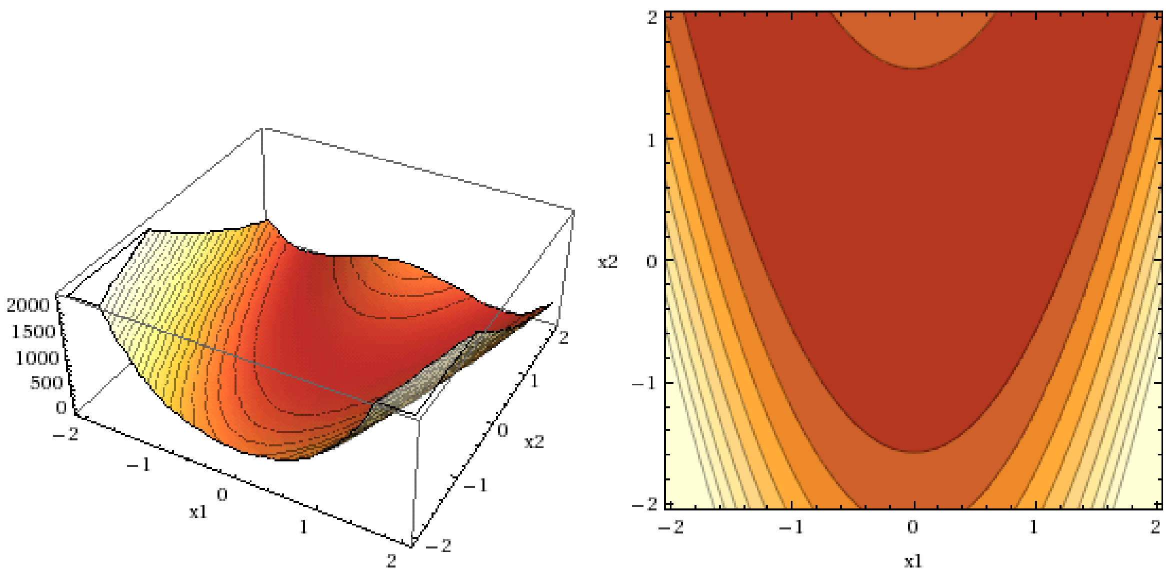

Figure 11: plot of test Rosenbrock's function used in this study.

Table 3: Comparison of algorithms COLM and ICOLM for f_2.

algo	Best value	Mean value	Std.Dev	$\begin{pmatrix}\bar{x}\\ \bar{y}\end{pmatrix}$
Griewank's Function				
IC1	0.0000	0.0072	0.0010	$\begin{pmatrix}-0.0003\\ 0.0003\\ -0.0011\end{pmatrix}$
C1	0.3775	0.3979	0.0080	$\begin{pmatrix}0.6106\\ 0.9361\\ 0.3953\end{pmatrix}$
IC2	0.0000	0.0067	0.0022	$\begin{pmatrix}-0.0060\\ -0.0018\\ -0.0091\end{pmatrix}$
C2	0.0218	0.0916	0.0402	$\begin{pmatrix}-0.0064\\ 0.2170\\ 0.0013\end{pmatrix}$
IC3	0.0074	0.0267	0.0238	$\begin{pmatrix}-0.1318\\ -0.0213\\ -0.0107\end{pmatrix}$
C3	0.0008	0.0130	0.0070	$\begin{pmatrix}-0.1705\\ -0.2763\\ -0.3607\end{pmatrix}$
IC4	0.0000	0.0000	0.0000	$\begin{pmatrix}-0.0040\\ -0.0067\\ -0.0060\end{pmatrix}$
C4	0.0011	0.0117	0.0070	$\begin{pmatrix}-0.1306\\ -0.2616\\ -0.2961\end{pmatrix}$

Table 4: Comparison of algorithms COLM and ICOLM for f_3.

algo	Best value	Mean value	Std.Dev	$\begin{pmatrix}\bar{x}\\ \bar{y}\end{pmatrix}$
Rosenbrock's Function				
IC1	0.0001	0.0106	0.0114	$\begin{pmatrix}0.9030\\ 0.8143\\ 0.6617\end{pmatrix}$
C1	0.0034	0.0060	0.0043	$\begin{pmatrix}1.0213\\ 1.0486\\ 1.1041\end{pmatrix}$
IC2	0.0000	0.0012	0.0010	$\begin{pmatrix}0.9700\\ 0.9413\\ 0.8840\end{pmatrix}$
C2	0.0003	0.0065	0.0039	$\begin{pmatrix}1.0066\\ 1.0060\\ 1.0135\end{pmatrix}$
IC3	0.0000	0.0032	0.0047	$\begin{pmatrix}0.9604\\ 0.9198\\ 0.8449\end{pmatrix}$
C3	0.0231	0.2153	0.0670	$\begin{pmatrix}1.0317\\ 1.0351\\ 1.1113\end{pmatrix}$
IC4	0.0000	0.0000	0.0000	$\begin{pmatrix}0.9968\\ 0.9937\\ 0.9880\end{pmatrix}$
C4	0.0143	0.1799	0.0884	$\begin{pmatrix}1.0317\\ 1.0351\\ 1.1113\end{pmatrix}$

6 Conclusion

In every test, with the same computational cost, ICOLM gives better results than COLM best values and Mean Best values but in one case. The presented study allows us to conclude that the proposed method is fast and converges to a good optimum. As we used a sampling mechanism to coordinate the research methods based on chaos theory, and we refined the final solution using a second method of local search. Further research is needed to gain more confidence and better understanding of the proposed methodology. The proposed algorithm has to be evaluated for a large number of test functions in higher dimension.

References

[1] L. S. Coelho. “Tuning of PID controller for an automatic regulator voltage system using chaotic optimization approach”, Chaos, Solitons and Fractals, 39, 2009, 1504-1514.

[2] L. S. Coelho. “Reliability-redundancy optimization by means of a chaotic differential evolution approach”, Chaos, Solitons and Fractals, 41, 2009, 594-602.

[3] D. Davendra, I. Zelinka, R. Senkerik, “Chaos driven evolutionary algorithms for the task of PID control”, Computers and Mathematics with Applications, 60, 2010, 1088-1104.

[4] R. Caponetto, L. Fortuna, S. Fazzino and M. G. Xibilia, “Chaotic Sequences to Improve the Performance of Evolutionary Algorithms”, IEEE Transactions on Evolutionary Computation, Vol. 7, n 3, June 2003, 289-304.

[5] T. Y. Li and J. A. Yorke, “Period three implies chaos”, Amer. Math. Monthly, 82 , 1975, 985-992.

[6] R. Lozi, “Un attracteur étrange? du type attracteur de Hénon”, J. Phys., 39 (C5), 1978, 9-10.

[7] S. H. Strogatz, “Nonlinear dynamics and chaos”, Massachussets: Perseus Publishing, 2000.

[8] T. S. Parker, L.O. Chua, “Practical numerical algorithms for chaotic system”. Berlin, Germany: Springer, 1989.

[9] K. T. Alligood, T. D. Sauer, J. A. Yorke, “Chaos: an introduction to dynamical systems”, London, UK: Springer; 1996.

[10] M. Hénon, "A two dimensional mapping with a strange attractor”, Commun. Math. Phys. 1976; 50(1):69-77.

[11] S. Cong, G. Li and X. Feng, “An Improved Algorithm of Chaos Optimization”, Proceedings 2010, 8th IEEE International Conference on Control and Automation.

[12] M.S. Tavazoei and M. Haeri, “Comparison of Different One-Dimensional Maps as Chaotic Search Pattern in Chaos Optimization Algorithms”, Applied Mathematics and Computation, Vol. 187, No. 2, pp. 1076-1085, 2007.

[13] M.S. Tavazoei and M. Haeri, "An optimization algorithm based on chaotic behavior and fractal nature", Journal of Computational and Applied Mathematics, Vol. 206, No. 2, 2007, pp.1070-1081.

[14] H. Shayeghi, S. Jalilzadeh, H.A. Shayanfar, A. Safari, "Robust PSS Design using Chaotic Optimization Algorithm for a Multimachine Power System". ECTI-CON 2009, Pattaya, Thailand, May 2009, pp. 40 - 43.

[15] H. Shayeghi, H. A. Shayanfar, S. Jalilzade, A. Safari, "A PSO based Unified Power Flow Controller for Damping of Power System Oscillations". Energy Conversion and Management, Vol.50, No. 10, Oct. 2009, pp. 2583 – 2592.